



# **A2019 Salesforce Multi Language Package**

## **Readme**

---

**Version 1.0**  
**25/01/2022**

## Table of Contents

<b>1. Introduction .....</b>	<b>3</b>
1.1 Overview .....	3
1.2 Use cases .....	3
<b>2. Requirements &amp; Prerequisites .....</b>	<b>3</b>
2.1 System Requirements.....	3
<b>3. Getting Started .....</b>	<b>4</b>
3.1 Quick Start .....	4
<b>3.1.1 Setup .....</b>	<b>4</b>
<b>3.1.1 Different actions under the Package .....</b>	<b>5</b>

# 1. Introduction

---

This document contains all essential information for the user to make full use of this A2019 Bot. It includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the Bot.

## 1.1 Overview

Use the A2019 Salesforce Multi Language Package to create, delete, update & run SOQL queries using Salesforce Org username, password , client secret & key details.

## 1.2 Use cases

The key use cases for this bot include:

- Salesforce Authentication.
- Salesforce Object creation like Case/Account/Opportunity.
- Salesforce SOQL query execution.
- Salesforce existing Object updation for Case/Account/Opportunity.
- Salesforce Object deletion like Case/Account/Opportunity.

# 2. Requirements & Prerequisites

## 2.1 System Requirements

Below are the minimum system requirements for running bots as an Enterprise A2019 (Cloud deployed) or Community Edition user on your local machine.

### Hardware Requirements

Device	Processor	RAM	Storage (Free Space)	Network
Machine	Intel Core i5 2.6 GHz	4 GB minimum 8 GB recommended	32 GB	1 GbE

Bot Creator and Bot Runner	No additions to the machine requirements	No additions to the machine requirements	Add 100 through 150 KB per Automation Anywhere script  Add 40 through 50 GB per long-term project	No additions to the machine requirements
----------------------------	------------------------------------------	------------------------------------------	---------------------------------------------------------------------------------------------------------	------------------------------------------

## Browser Requirements

The user interface for Automation Anywhere Enterprise is through a browser. Login to your device then login to Enterprise Control Room through a browser.

Browser	Browser Version	Automation Anywhere plug-in version <sup>2</sup>	Supported bot functions by Bot agent version		
Google Chrome <sup>1</sup>	57 or later	11 or 12	2.0.2	1.0.2	1.0.1
Microsoft Internet Explorer	11	N/A	None	Debugger only	All except Credential Vault

## 3. Getting Started

### 3.1 Quick Start

#### 3.1.1 Setup

Create your account in Salesforce Org <https://developer.salesforce.com> & then generate Client ID & Client Secret using steps 1-7.

1. Click on New Connected App in Salesforce screen.
2. Enter the name & the email address.
3. Click on Enable OAuth Settings, Enable for Device Flow & Use Digital Signatures.
4. Choose the certificate that you have just downloaded.
5. Under Available OAuth Scopes, select the below ones.
  - a. Access & manage your Data(api)
  - b. Perform request on your behalf at any time (refresh\_token, offline\_access)
6. Click on Save & then Continue.
7. Copy the Consumer ID & key from the screen.

## 3.1.1 Different actions under the Package

### Session Management

To use the Salesforce multi-language package, firstly need to create a session to authenticate your credentials.

Enter your credentials as shown in the below screenshot.

#### **Salesforce: Authenticate**

Authenticates via the Salesforce API to generate an Access Token

Session Name

☞ Default (x)

Base URL for your Salesforce Env

Credential Variable Insecure string

☞ \$salesforceOrgURL\$ (x)

Without trailing slash - Example: https://ap17.salesforce.com

Client ID

Credential Variable Insecure string

☞ \$client\_id\$ (x)

Client Secret

Credential Variable Insecure string

☞ \$client\_secret\$ (x)

Salesforce Username

Credential Variable Insecure string

☞ \$username\$ (x)

User must have access to authenticate via API

Salesforce Password

Credential Variable Insecure string

☞ \$password\$ (x)

Access Token Response

☞ Authenticate (x)

## Inserting Object in Salesforce

To insert Objects into Salesforce, use the below action in the package.

### **Salesforce: Insert Object**

Create Salesforce Objects

Session Name

Default (x)

Object Type for Updating

Case (x)

Example: Case, User, Opportunity, etc.

Value(s) to Insert (as key value pairs)

Dictionary Variable

\$dNewCase\$ (x)

Dictionary of fields to be updated where key is the Object Attribute name (OwnerId for example) and value is the intended update value.

Insert Object Response

sInsertedObjectID (x)

New Object ID is returned on success, error message on failure

## Executing SOQL Queries in Salesforce

To execute SOQL queries into Salesforce, use the below action in the package.

### **Salesforce: Execute SOQL**

Execute a SOQL Query and get a response as JSON

Session Name

Default (x)

SOQL Query to execute

SELECT CaseNumber,Priority,ContactEmail,Description,Id FROM Case WHE (x)

Query Response (JSON on success)

sQueryRunDetails (x)

## Updating existing Objects into Salesforce

To update an existing object into Salesforce, use the below action in the package.

### **Salesforce: Update Object**

Update Object by ObjectID and Type

Session Name

☞ Default

(x)

Object Type for Updating

☞ Case

(x)

Example: Case, User, Opportunity, etc.

ObjectID of object to update

☞ \$InsertedObjectID\$

(x)

Value(s) To update (as key value pairs)

Dictionary

Variable

☞ \$dNewCase\$

(x)

Dictionary of fields to be updated where key is the Object Attribute name (OwnerId for example) and value is the intended update value.

Update Object Response ('Object Updated Successfully' on success)

☞ sObjectUpdateResult

(x)

## Delete an existing Objects into Salesforce

To delete an existing object into Salesforce, use the below action in the package.

### **Salesforce: Delete Object**

Delete a Salesforce Object

Session Name

☞ Default

(x)

Object Type for Deletion

☞ Case

(x)

Example: Case, User, Opportunity, etc.

Object ID for object targetted for deletion

☞ \$InsertedObjectID\$

(x)

Delete Object Response

☞ sDeletedObjectResult

(x)

Successfully deleted message on success, error body on failure